TITLE OF THE INVENTION

Interleaving/deinterleaving method and apparatus

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a method and an apparatus for interleaving/deinterleaving transmission/reception data in bit units in the field of information communication.

2. Prior Art

Conventionally, an interleaving apparatus interleaved data in word units, since data was read from storage means in word units. However, when a burst error occurred, interleaving in word units had a defect of causing many errors in one word.

Hence, an apparatus for interleaving data in bit units has been proposed in Japanese Patent No. 2999101 (Japanese Laid-open Patent Application No. Hei 07-049779). This interleaving apparatus sequentially reads data elements before interleaving, interleaves the data, carries out a logical operation of the data and the data (interleaving intermediate data in the middle of the processing) stored in the storage area, in which interleaved data is to be stored, of the storage means, and then stores the interleaved data at the same address of the storage means. The interleaving apparatus carries out the above-mentioned processing, thereby attaining interleaving. The interleaving apparatus described in Japanese Patent No. 2999101 (Japanese Laid-open Patent Application No. Hei 07-049779) is hereafter taken as a conventional example, and its configuration and operation will be described in detail below by using FIGS. 13 and 14.

FIG. 13 is a schematic diagram showing the interleaving apparatus in accordance with the conventional example. Numeral 10 designates data storage means, numeral 20 designates access information supply means, numeral 30 designates a first pointer, numeral 60 designates a third pointer, numeral 200 designates a multiplexer, numeral 210 designates logical operation means, numeral 220 designates a 1-bit shifter, numeral 230 designates a register, numeral 240 designates a latch, and numeral 250 designates control means.

The reception data stored in the data storage means 10 is read in word units by the first pointer 30 in accordance with the sequence of addresses and stored in r1 of the register 230. At that time, the data element in the most significant bit is stored in the latch 240.

The access information supply means 20 stores address information and bit position information, respectively regarding addresses and bits in which interleaved data elements are stored, in address storage means 21 in accordance with the sequence of addresses obtained before interleaving. The address information is output to the third pointer 60, and 1-word data is read from the address designated by the third pointer 60. The bit position information is output to the control means 250. When the data element stored in the latch 240 is 0, the control means 250 outputs 1-word data wherein the value in the bit position designated by the bit position information is 0 and the values at the other bit positions are 1. The logical operation means 210 computes the AND of the 1-word data of the control means 250 with the 1-word data read from the address designated by the third pointer 60 and stores the AND in r0. On the other hand, when the data element stored in the latch 240 is 1, the control means 250 outputs 1-word data wherein the value in the bit position designated by the bit position information is 1 and the values at the other bits are 0. The logical operation means 210 computes the OR of the 1-word data of the control means 250 with the 1-word data read from the address designated by the third pointer 60 and stores the OR in r0. The data stored in r0 is written in the data storage means 10. Hence, among the data elements in the 1-word data stored at the address designated by the third pointer 60, only the data element in the bit corresponding to the bit position information is rewritten to the data element stored in the latch 240, whereby the data in the data storage means 10 is overwritten.

Then, the data in r1 is shifted to higher-order bits by one bit and stored again in r1, and the data element in the most significant bit is stored in the latch 240. The above-mentioned procedure is repeated. When the procedure is completed for all the data elements of the 1-word data stored in r1, 1-word data is read again from the address designated by the first pointer 30, and the above-mentioned procedure is repeated. Interleaving ends when the procedure is completed for all the reception data.

FIG. 14 is a table illustrating a program in accordance with the conventional example. At step 0, the address information supplied by the access information supply means 20 is set in the third pointer 60. At step 1, the 1-word data at the address designated by the first pointer 30 is stored in r1, the data element in the most significant bit is stored in the latch 240, and the first pointer 30 is incremented by one. At step 2, the number of repetitions is set at a number so that the processing is repeated by the number corresponding to the bit width of one word. At step 3, the 1-word data stored at the address, designated by the third pointer 60, of the data storage means 10 is read, the data element held in the latch 240 is set in the bit position designated by the bit position information, and the obtained data is stored in r0, and a second pointer 22 is incremented by one. At step 4, the 1-word data in r0 is transferred to the address, designated by the third pointer 60, of the data storage means 10. At step 5, the data in r1 is shifted to higher-order bits by one bit and stored in r1. The data element in the most significant bit is stored in the latch 240, the next address information is set in the third pointer 60. If the number of repetitions is not more than the preset number, the procedure returns to step 3. If the number of repetitions is more than the preset number, step 6 is then carried out. At step 6, if interleaving has not been carried out for all the data elements, the procedure returns to step 1. If interleaving has been carried out, the program ends. The program comprises the above-mentioned processing steps.

Deinterleaving is carried out in accordance with a procedure similar to that described above, since deinterleaving is different from interleaving only in the address information and the bit position information stored in the address storage means 21.

However, in the conventional example, the addresses and bit positions, in which interleaved data elements are stored, of the data storage means 10 are not continuous. Hence, in order to store one data element, it is necessary to carry out a logical operation with the interleaving intermediate data at the address of the storage destination. In addition, the processing by the logical operation means 210 must be switched depending on whether the data element is 0 or 1. This has a problem of making the size of the circuit larger and making the logical operation more complicated. Furthermore, it is necessary to write 1-word data in the data storage means 10 each time 1-bit data is interleaved. This has a problem of increasing the number of processing steps and the number of the total processing steps of the program.

Moreover, problems similar to those described above occur in the case of deinterleaving, since deinterleaving is different from interleaving only in the address information and the bit position information stored in the address storage means 21.

Still further, in order that selection is made between interleaving and deinterleaving in the conventional example, there is a problem of requiring that the address information and the bit position information are rewritten.

SUMMARY OF THE INVENTION

The present invention is intended to provide an interleaving/deinterleaving method and apparatus capable of avoiding complicated logical operation processing and of reducing the size of the circuit and the number of processing steps, and further intended to provide an interleaving/deinterleaving apparatus capable of making the rewriting of address information and bit position information for selection between interleaving and deinterleaving unnecessary.

A first interleaving/deinterleaving method in accordance with the present invention for interleaving/deinterleaving first data stored in data storage means to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises a first step of reading word data, part of the first data, from the data storage means, and a second step of selecting a data element to be processed from the word data read and of outputting the selected data element, the first step and the second step being repeated, wherein the sequence of the data elements to be processed at the time of the repetition is determined in accordance with the arrangement of the data elements of the second data.

A second interleaving/deinterleaving method in accordance with the present invention for interleaving/deinterleaving first data stored in data storage means to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data in bit units, comprises a first step of reading word data, part of the first data, from the data storage means, a second step of selecting a data element to be processed from the word data read and of outputting the selected data element, a third step of shifting a bit sequence having already been stored in a shift register by one bit and of storing the data element selected and output in the shift register, and a fourth step of storing the bit sequence stored in the shift register in the data storage means after the processing from the first step to the third step is repeated by a predetermined number of times, wherein the sequence of the data elements to be processed at the time of the repetition of the processing from the first step to the third step is determined in accordance with the arrangement of the data elements of the second data.

In the above-mentioned first and second interleaving/deinterleaving methods, interleaving/deinterleaving is carried out in units of data elements in sequence in consideration of the arrangement of data elements after interleaving/deinterleaving instead of before interleaving/deinterleaving. With this method, the processed data can be output sequentially without being further processed at all. Hence, logical operation processing required in the conventional example becomes unnecessary. It is thus possible to reduce the size of the circuit, the number of the processing steps and the number of the total processing steps of the program.

In the second interleaving/deinterleaving method, in order that the processed second data is stored once in the data storage means, a shift register having the data width of at least one word is provided, and after the data elements of one word are stored in the shift register, the data elements are written in the data storage means.

A third interleaving/deinterleaving method in accordance with the present invention for interleaving/deinterleaving first data stored in data storage means to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises a step of initializing an area, in which the second data is stored, of the data storage means, a step of reading word data, part of the first data, from the data storage means in accordance with the arrangement of the data elements of the first data and of storing the word data in a shift register, a step of sequentially shifting out the word data stored in the shift register beginning with the data element positioned in the most significant bit position, a step of generating first word data by positioning the data element shifted out at a bit position in word data serving as the second data and by expanding the data element to word data, a step of reading as second word data the word data stored at the address, in which the data element shifted out should be stored, of the data storage means, a step of computing the OR of the first word data with the second word data, and a step of storing the OR at the address, from which the second word data is read, in the data storage means.

In the third interleaving/deinterleaving method, interleaving/deinterleaving is carried out in the sequence of the data elements before interleaving/deinterleaving just as in the case of the conventional example. The address information and the bit position information supplied by access information supply means designate the address and the bit position in which the word data serving as the second data should be stored. However, by previously initializing an area in which interleaved/deinterleaved data should be stored, it is possible to uniformly "set 0 at bit positions other than the bit position to which the corresponding data element is output" as a data expansion method. Furthermore, logical operation processing can be limited to logical OR processing. Hence, in comparison with the conventional example, logical operation processing can be simplified, the size of the circuit can be reduced, and the total processing steps can be reduced.

In addition, a first interleaving/deinterleaving apparatus in accordance with the present invention for interleaving/deinterleaving first data to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises data storage means for storing the first data, access information supply means for sequentially supplying the address information of word data, part of the first data, in the data storage means and the bit position information of the data element to be processed in the word data, and data selection means for receiving the word data read from the address, corresponding to the address information, of the data storage means, for selecting one data element from the word data on the basis of the bit position information and for outputting the data element, wherein the address information and the bit position information supplied sequentially by the access information supply means are determined in accordance with the arrangement of the data elements of the second data.

The first interleaving/deinterleaving apparatus may be configured so as to be characterized in that the data elements are processed in bit units, that a shift register for sequentially shifting, by one bit, a bit sequence having already been stored and for sequentially storing a 1-bit data element sequentially output from the data selection means is further provided, and that when the bit width of the bit sequence stored in the shift register reaches a predetermined bit width, not smaller than the bit width of at least 1-word data, the bit sequence is stored as part of the second data in the data storage means.

The first interleaving/deinterleaving apparatus may be configured so as to be characterized in that the data selection means comprises a shifter, that the word data read from the data storage means is shifted so that the data element to be selected is positioned at a specific bit position, and that the output from the specific bit position after the shifting is used as the output of the data selection means.

In the first interleaving/deinterleaving apparatus described above, interleaving/deinterleaving is carried out in units of data elements in sequence in consideration of the arrangement of data elements after interleaving/deinterleaving instead of before interleaving/deinterleaving. With this method, the processed data can be output sequentially without being further processed at all. Hence, logical operation processing required in the conventional example becomes unnecessary. It is thus possible to reduce the size of the circuit, the number of the processing steps and the number of the total processing steps of the program. The sequence of data elements to be interleaved/deinterleaved should only be held in the access information supply means for supplying address information designating the address, at which word data, part of the first data, is stored, in the data selection means and bit position information designating a bit position in the word data.

Furthermore, in order that the processed second data is stored once in the data storage means, a shift register having the bit width of at least one word should only be provided, and after the data elements of one word are stored in the shift register, the data elements should only be written in the data storage means.

A second interleaving/deinterleaving apparatus in accordance with the present invention for interleaving/deinterleaving first data to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises data storage means for storing the first data and the second data, access information supply means for supplying the address information of word data, part of the second data, in the data storage means and the bit position information of the data element to be processed in the word data, a shift register for storing word data, part of the first data, read from the data storage means in accordance with the arrangement of the data elements of the first data and for sequentially shifting out the word data beginning with the data element positioned in the most significant bit position, data expansion means for sequentially expanding the data element shifted out sequentially from the shift register on the basis of the bit position information to word data and for outputting the word data, and logical OR means for computing the OR of the word data read from the address, corresponding to the address information, of the data storage means with the word data output from the data expansion means and for outputting the OR, wherein the OR output from the logical OR means is stored at the address, corresponding to the address information, from which the word data is read, of the data storage means, and the address information and the bit position information supplied sequentially by the access information supply means are determined in accordance with the arrangement of the data elements of the first data.

The second interleaving/deinterleaving apparatus may be configured so as to be characterized in that the data expansion means comprises a shifter, that the data element shifted out from the shift register is shifted so as to be positioned in the bit position corresponding to the bit position information, that 0 is set in the bits higher and lower than the bit position corresponding to the bit position information, and that the obtained data is output as word data.

In the above-mentioned second interleaving/deinterleaving apparatus, interleaving/deinterleaving is carried out in the sequence of the data elements before interleaving/deinterleaving just as in the case of the conventional example. The address information and the bit position information supplied by access information supply means designate the address and the bit position in which the word data serving as the second data should be stored. However, by previously initializing an area in which interleaved/deinterleaved data should be stored, it is possible to uniformly "set 0 at bit positions other than the bit position to which the corresponding data element is output" as a data expansion method. Furthermore, logical operation processing can be limited to logical OR processing. Hence, in comparison with the conventional example, logical operation processing can be simplified, the size of the circuit can be reduced, and the total processing steps can be reduced.

A third interleaving/deinterleaving apparatus in accordance with the present invention for interleaving/deinterleaving first data to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises data storage means for storing the first data and the second data, and access information supply means for supplying address information and bit position information, wherein the address information designates the address of word data, part of the second data, in the data storage means at the time of interleaving or designates the address of word data, part of the first data, in the data storage means at the time of deinterleaving, and the bit position information designates the bit position of the data element to be processed in the word data, part of the second data, at the time of interleaving or designates the bit position of the data element to be processed in the word data, part of the first data, at the time of deinterleaving.

The third interleaving/deinterleaving apparatus may comprise a shift register for storing word data, part of the first data, read from the data storage means in accordance with the arrangement of the data elements of the first data and for sequentially shifting out the word data beginning with the data element positioned in the most significant bit at the time of interleaving, data selection expansion means for sequentially expanding the data element shifted out sequentially from the shift register on the basis of the bit position information to word data and for outputting the word data at the time of interleaving, and logical OR means for computing the OR of the word data read from the address, corresponding to the address information, of the data expansion means with the word data output from the data selection expansion means and for outputting the OR at the time of interleaving, wherein the OR output from the logical OR means is stored at the address, corresponding to the address information, from which the word data is read, of the data expansion means, the data selection expansion means receives the word data read from the address, corresponding to the address information, of the data expansion means, selects one data element from the word data on the basis of the bit position information and outputs the data element at the time of deinterleaving, and the shift register sequentially shifts, by one bit, a bit sequence having already been stored and sequentially stores a 1-bit data element sequentially output from the data selection expansion means at the time of deinterleaving; when the bit width of the bit sequence stored in the shift register reaches a predetermined bit width, not smaller than the bit width of at least 1-word data, the bit sequence is stored as part of the second data in the data storage means.

A fourth interleaving/deinterleaving apparatus in accordance with the present invention for interleaving/deinterleaving first data to second data so that the arrangement of the data elements of the second data is different from the arrangement of the data elements of the first data, comprises data storage means for storing the first data and the second data, and access information supply means for supplying address information and bit position information, wherein the address information designates the address of word data, part of the second data, in the data storage means at the time of deinterleaving or designates the address of word data, part of the first data, in the data storage means at the time of interleaving, and the bit position information designates the bit position of the data element to be processed in word data, part of the second data, at the time of deinterleaving or designates the bit position of the data element to be processed in word data, part of the first data, at the time of interleaving.

The fourth interleaving/deinterleaving apparatus may comprise a shift register for storing word data, part of the first data, read from the data storage means in accordance with the arrangement of the data elements of the first data and for sequentially shifting out the word data beginning with the data element positioned in the most significant bit at the time of deinterleaving, data selection expansion means for sequentially expanding the data element shifted out sequentially from the shift register on the basis of the bit position information to word data and for outputting the word data at the time of deinterleaving, and logical OR means for computing the OR of the word data read from the address, corresponding to the address information, of the data expansion means with the word data output from the data selection expansion means and for outputting the OR at the time of deinterleaving, wherein the OR output from the logical OR means is stored at the address, corresponding to the address information, from which the word data is read, of the data expansion means, the data selection expansion means receives the word data read from the address, corresponding to the address information, of the data expansion means, selects one data element from the word data on the basis of the bit position information and outputs the data element at the time of interleaving, and the shift register sequentially shifts, by one bit, a bit sequence having already been stored and sequentially stores a 1-bit data element sequentially output from the data selection expansion means at the time of interleaving; when the bit width of the bit sequence stored in the shift register reaches a predetermined bit width, not smaller than the bit width of at least 1-word data, the bit sequence is stored as part of the second data in the data storage means.

The above-mentioned third and fourth interleaving/deinterleaving apparatuses are configured so that the components of the first and second interleaving/deinterleaving apparatuses described before are used in common. There is a reverse conversion relationship between interleaving and deinterleaving with respect to the placement of data elements. Hence, in the case when the same address information and bit position information are provided for the access information supply means of the first and second interleaving/deinterleaving apparatuses, one of the apparatuses can carry out interleaving and the other can carry out deinterleaving. In the conventional example and the first and second interleaving/deinterleaving apparatuses, in order that selection is made between interleaving and deinterleaving, it is necessary to rewrite the address information and the bit position information. However, in the third and fourth interleaving/deinterleaving apparatuses, such rewriting is not necessary.

BRIEF DESCRIPTION OF DRAWINGS

FIG. 1 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a first example of a first embodiment of the present invention;

FIG. 2 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a second example of the first embodiment of the present invention;

FIG. 3 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a second embodiment of the present invention;

FIG. 4 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a third embodiment of the present invention;

FIG. 5 is a table illustrating the placement of data elements at the time of deinterleaving in accordance with the first embodiment of the present invention;

FIG. 6 is a table illustrating the storage information of address storage means at the time of deinterleaving in accordance with the first embodiment of the present invention;

FIG. 7 is a table illustrating the placement of data elements at the time of interleaving in accordance with the first embodiment of the present invention;

FIG. 8 is a table illustrating the storage information of the address storage means at the time of interleaving in accordance with the first embodiment of the present invention;

FIG. 9 is a table illustrating the storage information of address storage means at the time of deinterleaving in accordance with the second embodiment of the present invention;

FIG. 10 is a table illustrating the storage information of the address storage means at the time of interleaving in accordance with the second embodiment of the present invention;

FIG. 11 is a table illustrating a program in accordance with the first embodiment of the present invention;

FIG. 12 is a table illustrating a program in accordance with the second embodiment of the present invention;

FIG. 13 is a schematic diagram showing the configuration of the conventional example; and

FIG. 14 is a table illustrating a program in accordance with the conventional example.

DESCRIPTION OF PREFERRED EMBODIMENTS

Embodiments in accordance with the present invention will be described below referring to the drawings.

(First embodiment)

FIG. 1 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a first embodiment. For the sake of simplicity, explanations are first given on the assumption that deinterleaving is carried out.

A data storage means 10 holds interleaved reception data. Read/write access is possible at the address designated by a first pointer 30.

An access information supply means 20 comprises address storage means 21 and a second pointer 22. The address storage means 21 holds address information and bit position information for access to the elements of reception data in sequence in consideration of the arrangement of the data elements of deinterleaved data in accordance with the sequence of the addresses of the address storage means 21. The address storage means 21 outputs the address information and the bit position information stored at the address designated by the second pointer 22. The second pointer 22 first designates the head address of the address storage means 21 and is incremented by one each time processing is carried out. In this way, the access information supply means 20 outputs the address information of the data storage means 10 to the first pointer 30 and outputs the bit position information to data selection means 40 described later.

The data selection means 40 selects the data element in the bit position designated by the bit position information from 1-word data read from the data storage means 10 and outputs the data element. The bit sequence continuously processed in this way has already become a deinterleaved data sequence and is directly supplied to data decoding means 100 provided externally.

FIG. 2 is a schematic diagram in accordance with the first embodiment provided with a function of storing deinterleaved data in the data storage means 10. In the case when the above-mentioned data decoding means 100 cannot process deinterleaved data in time sequence because of operation timing, processing procedure, etc., the data is required to be stored in the data storage means once. A shift register 50 shifts the data elements, having already been stored, to higher-order bits by one bit and stores a new data element output from the data selection means 40 in the least significant bit. After 1-word data is stored by the repetition of this processing, the data is output to the data storage means 10. The third pointer 60 first designates the head address of the storage area in which deinterleaved data is stored. When the 1-word data output from the shift register 50 is stored at the corresponding address in the data storage means 10, the address information having been held is incremented by one and thus renewed to designate the next address.

FIG. 5 is a table illustrating the placement of data elements stored in the data storage means 10 before and after processing at the time of deinterleaving. For the sake of simplicity, it is herein assumed that the bit width of one word is eight bits, that the data width subjected to deinterleaving is 32 bits, and that reception data has already been interleaved by block interleaving wherein the column direction of the data is converted into the row direction. Hence, deinterleaving is carried out by converting the row direction of the data into the column direction. In addition, it is assumed that the reception data is stored at address 100 and the subsequent addresses in the data storage means 10, that deinterleaved data is stored at address 300 and the subsequent addresses, and that the data elements of the reception data are d0, d1, d2, ..., d31 in sequence, beginning with the head thereof. Furthermore, in actual storage means, plural words (plural addresses) are arranged physically in the row direction. However, one row is used for one word (one address) for the sake of simplicity.

In the case when the data elements in the sequence of d0, d1, d2, ... before deinterleaving are deinterleaved, their respective storage destinations are bit 7 of address 300, bit 7 of address 301, bit 7 of address 302, bit 7 of address 303, ... of the data storage means 10. It is thus understood that preprocessing is required to write data to the data storage means 10 in word units just as in the case of the conventional example.

When the sequence of the data elements to be deinterleaved is d0, d4, d8, ... in consideration of the arrangement after the processing and the data elements are deinterleaved, the deinterleaved data elements can be directly supplied to the data decoding means 100 in the case of FIG. 1. Even in the case of storing the data elements in the data storage means 10 in the case of FIG. 2, their respective storage destinations are bit 7 of address 300, bit 6 of address 300, bit 5 of address 300, ... It is thus understood that after the data elements of 1-word data are stored in the shift register 50 in sequence, the 1-word data should only be written at address 300 of the data storage means 10.

FIG. 6 is a table illustrating the address information and the bit position information held in the address storage means 21 constituting the access information supply means 20 at the time of deinterleaving. The second pointer 22 first designates address 0 of the address storage means 21. The access information supply means 20 outputs 100 as the address information of the data storage means 10 to the first pointer 30 and outputs bit 7 as the bit position information. The second pointer 22 is updated so as to designate the next address. In this way, the first pointer 30 designates address 100, and the 1-word data [d0, d1, d2, d3, d4, d5, d6, d7] stored at address 100 is read from the data storage means 10. The data selection means 40 refers to bit 7 serving as bit position information and selects and outputs d0. After this, similarly, deinterleaved data elements are output in the sequence of d4, d8, d12, ... in accordance with the address information and the bit position information stored in the address storage means 21. In order that these data elements are stored in the data storage means 10, after the data elements of one word are stored in the shift register 50, the data elements should only be written in the data storage means 10. At that time, the third pointer 60 first designates address 300 of the data storage means 10. After the data is written at address 300, the third pointer 60 is updated so as to designate address 301.

The data selection means 40 may be configured by using a shifter, wherein the 1-word data read from the data storage means 10 is shifted toward the least significant bit by the bit position value designated by the bit position information so that the necessary data element is always positioned in the least significant bit of output data, and the least significant bit is supplied to the data decoding means 100 or the shift register 50.

In the first embodiment, deinterleaving can be carried out as described above. Since interleaving is the reverse of deinterleaving and is different from deinterleaving only in address information and bit position information, interleaving can be carried out by using a similar procedure.

FIG. 7 is a table illustrating the placement of data elements stored in the data storage means 10 before and after processing at the time of interleaving. Data elements before interleaving are written at address 100 and the subsequent addresses, and data elements after interleaving are written at address 300 and the subsequent addresses. Since interleaving is carried out, the column direction of data is converted into the row direction.

FIG. 8 is a table illustrating the address information and the bit position information held in the address storage means 21 constituting the access information supply means 20 at the time of interleaving. The address information and the bit position information are determined in consideration of the arrangement of interleaved data elements.

The access information supply means 20 herein comprises the address storage means 21 and the second pointer 22. However, since both the address information and the bit position information are changed in accordance with predetermined rules as shown in FIGS. 6 and 8, the address information and the bit position information may be generated by counting the number of processing for data elements.

FIG. 11 is a table illustrating a program in accordance with the first embodiment. At step 0, the address information supplied by the access information supply means 20 is set at the first pointer 30, and the second pointer 22 is incremented by one.

At step 1, the number of repetitions is set at a preset number so that the processing is repeated by the number corresponding to the bit width of one word.

At step 2, the 1-word data stored at the address, designated by the third pointer 60, of the data storage means 10 is read, the data element in the bit position designated by the bit position information is selected by the data selection means 40 and output to the shift register 50. The next address information supplied by the access information supply means 20 is set at the first pointer 30, and the second pointer 22 is incremented by one. If the number of repetitions is not more than the preset number, the procedure returns to the start of step 3. If the number of repetitions is more than the preset number, the next step, step 3, is then carried out.

At step 3, the 1-word data of the shift register 50 is transferred to the address, designated by the third pointer 60, of the data storage means 10, and the third pointer 60 is incremented by one.

At step 4, if deinterleaving/interleaving has not been carried out for all the data elements, the procedure returns to step 1. If deinterleaving/interleaving has been carried out, the program ends. The program comprises the above-mentioned processing steps.

FIG. 11 illustrates the program in the case of the configuration shown in FIG. 2. However, in the case when the third pointer 60 and the shift register 50 are not provided and when the output data from the data selection means 40 is not written in the data storage means 10, only step 3 in FIG. 11 is not required, whereby step 2 goes to step 4. However, in this case, the data element selected by the data selection means 40 at step 2 is output to the outside (the data decoding means 100 in the example shown in FIG. 1) instead of the shift register 50.

When FIG. 11 is compared with FIG. 14 serving as the table illustrating the program in accordance with the conventional example, it is found that the number of steps is fewer by two. Furthermore, in the first embodiment, step 2 is repeated by the number of times corresponding to the bit width of one word when 1-word data is interleaved/deinterleaved. However, in the conventional example, the processing in the sequence of step 3, step 4 and step 5 is repeated. Hence, it is found that the total number of processing steps can be further reduced in proportion to the size of data to be processed.

As described above, in the first embodiment, by devising a data accessing method, logical operation means is made unnecessary, unlike the case of the conventional example. It is thus possible to reduce the size of the circuit, the number of the processing steps and the number of the total processing steps of the program.

(Second embodiment)

FIG. 3 is a schematic diagram showing an interleaving/deinterleaving apparatus in accordance with a second embodiment. For the sake of simplicity, explanations are first given on the assumption that deinterleaving is carried out, just as in the case of the first embodiment.

A data storage means 10 holds interleaved reception data. The storage area in which interleaved data is stored is initialized by value 0. Read/write access is possible at the addresses designated by a first pointer 30 and a third pointer 60.

An access information supply means 20 comprises address storage means 21 and a second pointer 22. The address storage means 21 holds address information and bit position information for storage of processed data elements in sequence in consideration of the arrangement of the data elements of reception data in accordance with the sequence of the addresses of the address storage means 21. The address storage means 21 outputs the address information and the bit position information stored at the address designated by the second pointer 22. The second pointer 22 first designates the head address of the address storage means 21 and is incremented by one each time processing is carried out. In this way, the access information supply means 20 outputs the address information of the data storage means 10 to the third pointer 60 and outputs the bit position information to data expansion means 70 described later.

The 1-word data of the reception data read from the address designated by the first pointer 30 is stored in a shift register 50. The shift register 50 supplies the data elements to the data expansion means 70 while shifting the stored data to higher-order bits, one bit at a time.

The data expansion means 70 outputs 1-word data wherein the data element output from the shift register 50 is placed in the bit position designated by the bit position information and value 0 is placed at the other bit positions.

Logical OR means 80 computes the OR of the 1-word data read from the address designated by the third pointer 60 with the 1-word data output from the data expansion means 70. The data storage means 10 stores the 1-word data output from the logical OR means 80 at the address designated by the third pointer 60.

FIG. 9 is a table illustrating the address information and the bit position information held in the address storage means 21 constituting the access information supply means 20 at the time of deinterleaving. The second pointer 22 first designates address 0 of the address storage means 21. The access information supply means 20 outputs 300 as the address information of the data storage means 10 to the third pointer 60 and outputs bit 7 as the bit position information. The second pointer 22 is updated so as to designate the next address. The first pointer 30 first designates address 100 and is updated so as to designate address 101 after the 1-word data [d0, d1, d2, d3, d4, d5, d6, d7] stored at address 100 (see FIG. 5) of the data storage means 10 is read. The 1-word data read from address 100 is stored in the shift register 50, and the shift register 50 shifts out the data, whereby the data is supplied to the data expansion means 70 in the sequence of d0, d1, d2, ...

The data expansion means 70 generates and outputs 1-word data [d0, 0, 0, 0, 0, 0, 0, 0] wherein data element d0 is placed at bit 7 in accordance with data element d0 and bit 7 of the bit position information supplied and value 0 is placed at the other bit positions. The third pointer 60 designates address 300 on the basis of the address information. Initialized 1-word data [0, 0, 0, 0, 0, 0, 0, 0] stored at address 300 is read from the data storage means 10. The logical OR means 80 computes the OR of the 1-word data [0, 0, 0, 0, 0, 0, 0, 0] with the 1-word data [d0, 0, 0, 0, 0, 0, 0, 0] and outputs 1-word data [d0, 0, 0, 0, 0, 0, 0, 0]. The data storage means 10 stores the data output from the logical OR means 80 at address 300 designated by the third pointer 60. The third pointer 60 is updated so as to designate address 301 on the basis of the next address information.

The data expansion means 70 may be configured by using a shifter, wherein the data element supplied from the shift register 50 is shifted toward the most significant bit by the bit position value designated by the bit position information, the bits higher than the bit position of the data element are 0-expanded (0 is set at the higher-order bits), the lower-order bits are 0-padded (0 is set at the lower-order bits), and the obtained data is output as 1-word data.

In the second embodiment, deinterleaving can be carried out as described above. In addition, since interleaving is different from deinterleaving only in address information and bit position information, interleaving can be carried out by using a similar procedure.

FIG. 10 is a table illustrating the address information and the bit position information held in the address storage means 21 constituting the access information supply means 20 at the time of interleaving in the second embodiment.

The access information supply means 20 herein comprises the address storage means 21 and the second pointer 22. However, since both the address information and the bit position information are changed in accordance with predetermined rules as shown in FIGS. 9 and 10, the address information and the bit position information may be generated by counting the number of processing for data elements.

FIG. 12 is a table illustrating a program in accordance with the second embodiment. The storage area, in which processed data is stored, of the data storage means 10, hereafter referred to as a processed data storage area, is initialized at step 0 and step 1.

At step 0, the number of repetitions for the initialization is set at a preset number (1) so that all the processed data storage areas are initialized, and the address information supplied by the access information supply means 20 is set in the third pointer 60. The address information to be set at this time is the head address of the processed data storage area.

At step 1, value 0 is stored at the address (1-word data storage area), designated by the third pointer 60, of the data storage means 10, and the third pointer 60 is incremented by one. If the number of repetitions is not more than the preset number (1), the procedure returns to the start of step 1. If the number of repetitions is more than the preset number (1), the processing at the next step, step 2, is carried out.

At step 2, the address information supplied by the access information supply means 20 is set at the third pointer 60, and the second pointer 22 is incremented by one.

At step 3, the 1-word data stored at the address, designated by the first pointer 30, of the data storage means 10 is read and stored in the shift register 50, and the first pointer 30 is incremented by one.

At step 4, the number of repetitions is set at a preset number (2) so that the processing is repeated by the number corresponding to the bit width of one word.

At step 5, the logical OR means 80 computes the OR of the 1-word data read from the address, designated by the third pointer 60, of the data storage means 10 with the 1-word data output from the data expansion means 70. The OR is stored at the address, designated by the third pointer 60, of the data storage means 10. This stored data output from the logical OR means 80 is data obtained by setting the 1-bit output value of the shift register 50 in the bit position that is included in the 1-word data (all the data elements are 0) read from the address, designated by the third pointer 60, of the data storage means 10 and is designated by the bit position information. Furthermore, the next address information supplied by the access information supply means 20 is set at the third pointer 60, and the second pointer 22 is incremented by one. If the number of repetitions is not more than the preset number (2), the procedure returns to the start of step 5. If the number of repetitions is more than the preset number (2), the processing at the next step, step 6, is carried out.

At step 6, if deinterleaving/interleaving has not been carried out for all the data elements, the procedure returns to step 3. If deinterleaving/interleaving has been carried out, the program ends. The program comprises the above-mentioned processing steps.

When FIG. 12 is compared with FIG. 14 serving as the table illustrating the program in accordance with the conventional example, it is found that the number of processing steps is the same. In the following description, FIG. 12 is compared with FIG. 14 with respect to the number of the total processing steps. First, in the second embodiment, initialization steps are required. Since the total number of data elements divided by the number of bits in one word is the number of repetitions, the number of initialization steps is "2 × 32/8 = 8" in this case. On the other hand, the number of repetitions of a step of processing each data element directly becomes the total number of data elements. This number is 32 in the second embodiment and "3 × 32 = 96" in the conventional example. Hence, even if the number of processing steps is increased at the initialization steps, the effect of decreasing the number of steps for processing each data element is more significant. It is thus found that the number of total steps can be made smaller than that in the conventional example.

In the second embodiment, the storage area in which processed data is stored is initialized by value 0 beforehand as described above. Hence, regardless of whether the data element to be processed is 0 or 1, data expansion can be carried out by the same procedure, and logical operation can be limited to logical OR operation. Therefore, in comparison with the conventional example, logical operation means and the like can be simplified, and the size of the circuit can be reduced. Furthermore, the number of total processing steps can be reduced, although the effect of the reduction is less significant than that of the first embodiment since the initialization steps for the data storage means 10 are required.

(Third embodiment)

FIG. 4 is a schematic diagram showing a deinterleaving/interleaving apparatus in accordance with a third embodiment. This embodiment is configured so that the components of the first and second embodiments are used in common. Data selection expansion means 90 is configured so as to have both the function of the data selection means 40 in accordance with the first embodiment and the function of the data expansion means 70 in accordance with the second embodiment. Furthermore, the shift register 50 shown in FIG. 4 is configured so as to have both the function of the shift register 50 shown in FIG. 2 and the function of the shift register 50 shown in FIG. 3. Still further, in this embodiment, the same area (address range) is used as an area (address range) in data storage means 10 for storing data before interleaving at the time of interleaving and also used as an area (address range) in the data storage means 10 for storing data after deinterleaving at the time of deinterleaving.

When interleaving is carried out, the 1-word data corresponding to the address information supplied from an access information supply means 20 via a first pointer 30 is read from the data storage means 10 and sent to the data selection expansion means 90 just as in the case of the first embodiment. At the data selection expansion means 90, the bit corresponding to the bit position information supplied from the access information supply means 20 is selected, and the selected bit is stored in the least significant bit of the shift register 50. The second pointer 22 is then incremented, and the next bit in consideration of the arrangement after interleaving is processed. The shift register 50 shifts its data by one bit and stores the next bit in the least significant bit. After the 1-word data is interleaved, the data is stored at the address, designated by the third pointer 60, of the data storage means 10, and the next word data is processed. In this case, the third pointer 60 is controlled by means not shown just as in the case shown in FIG. 2.

Furthermore, when deinterleaving is carried out, the 1-word data at the address designated by the first pointer 30 is read from the data storage means 10 and sent to the shift register 50 just as in the case of the second embodiment. The shift register 50 shifts out the 1-bit data and supplies the data to the data selection expansion means 90. The data selection expansion means 90 shifts the data to higher-order bits by the bit position value designated by the bit position information of the access information supply means 20. The bits higher than the corresponding bit position are 0-expanded, the lower-order bits are 0-padded, and the obtained data is output as 1-word data to the logical OR means 80. The logical OR means 80 computes the OR of the 1-word data output from the data selection expansion means 90 with the 1-word data at the address, designated by the third pointer 60, of the data storage means 10. The OR is written at the address, designated by the third pointer 60, of the data storage means 10. In this case, the first pointer 30 is controlled by means not shown just as in the case shown in FIG. 3. Furthermore, the third pointer 60 designates the address in accordance with the address information supplied from the access information supply means 20 just as in the case shown in FIG. 3.

Owing to the above-mentioned configuration, the data selection expansion means 90 can attain both the function of the data selection means 40 and the function of the data expansion means 70, and the shift register 50 shown in FIG. 4 can attain both the function of the shift register 50 shown in FIG. 2 and the function of the shift register 50 shown in FIG. 3. Hence, the third embodiment can carry out both the processing in the first embodiment and the processing in the second embodiment described above.

When FIG. 6 showing the address information and the bit position information stored in the address storage means 21 at the time of deinterleaving in the first embodiment is compared with FIG. 10 showing the address information and the bit position information stored in the address storage means 21 at the time of interleaving in the second embodiment, the bit position information has the same setting, although there is a difference between the first embodiment and the second embodiment in that data is read from addresses 100s in the first embodiment and that data is written to addresses 300s in the second embodiment. A similar matter can also be said when FIG. 8 is compared with FIG. 9.

In other words, in this embodiment, the same area (address range) is used as the data reading destination at the time of interleaving and also used as the data writing destination at the time of deinterleaving. Furthermore, interleaving is carried out by using processing similar to that of the first embodiment, and deinterleaving is carried out by using processing similar to that of the second embodiment. In the cases of the conventional example and the above-mentioned first and second embodiments, two kinds of address information for interleaving and deinterleaving are stored in the address storage means and selectively used depending on whether the processing is interleaving or deinterleaving. In the cases of this embodiment, however, the selection depending on whether the processing is interleaving or deinterleaving is not required. Therefore, the amount of information to be stored in the address storage means can be decreased, whereby the size of the circuit can be reduced.

Since the same area is used as the data reading destination at the time of interleaving and also used as the data writing destination at the time of deinterleaving in the third embodiment, no problem occurs when interleaving or deinterleaving is carried out individually. However, interleaving and deinterleaving cannot be carried out simultaneously. In order that this problem is solved, address information may be offset only when deinterleaving is carried out.

For example, it is assumed that the address storage means in accordance with the third embodiment stores the address information and the bit position information shown in FIG. 8. When interleaving is carried out, data to be interleaving is read from addresses 100s. When deinterleaving is carried out, deinterleaved data is written at the same addresses, i.e., addresses 100s. These cannot be carried out simultaneously as a matter of course. However, by using a configuration wherein the address information is offset by the amount of 200 addresses at the time of deinterleaving, deinterleaved data is written at addresses 300s, whereby interleaving and deinterleaving can be carried out simultaneously. Even in this case, since the address information is only offset, the size of the circuit is made smaller than that of the circuit for storing two kinds of address information for interleaving and deinterleaving. It is needless to say that the third embodiment may be configured so that the address information is offset only at the time of interleaving.

The present invention is not limited to the above-mentioned embodiments. For example, a general-purpose processor may also be used for processing, instead of the interleaving/deinterleaving apparatuses in accordance with the above-mentioned embodiments. Even in that case, an effect of decreasing the number of processing steps can be obtained as described referring to FIGS. 11 and 12.